)

ABSTRACT

Estimating program worst case execution time (WCET) is an important problem in the domain of real-time systems and embedded systems that are deadline-centric. If WCET of a program is found to exceed the deadline, it is either recoded or the target architecture is modified to meet the deadline. Predominantly, there exist three broad approaches to estimate WCET- static WCET analysis, hybrid measurement based analysis and statistical WCET analysis. Though measurement based analyzers benefit from knowledge of run-time behavior, amount of instrumentation remains a concern.

This thesis proposes a CPI-centric WCET analyzer that estimates WCET as a product of worst case instruction count (IC) estimated using static analysis and worst case cycles per instruction (CPI) computed using a function of measured CPI. In many programs, it is observed that IC and CPI values are correlated. Five different kinds of correlation are found. This correlation enables us to optimize WCET from the product of worst case IC and worst case CPI to a product of worst case IC and corresponding CPI. A prime advantage of viewing time in terms of CPI, enables us to make use of program phase behavior. In many programs, CPI varies in phases during execution. Within each phase, the variation is homogeneous and lies within a few percent of the mean. Coefficient of variation of CPI across phases is much greater than within a phase. Using this observation, we estimate program WCET in terms of its phases. Due to the nature of variation of CPI within a phase in such programs, we can use a simple probabilistic inequality- Chebyshev inequality, to compute bounds of CPI within a desired probability. In some programs that execute many paths depending on if-conditions, CPI variation is observed to be high. The thesis proposes a PC signature that is a low cost way of profiling path information which is used to isolate points of high CPI variation and divides a phase into smaller sub-phases of lower CPI variation. Chebyshev inequality is applied to sub-phases resulting in much tighter bounds. Provision to divide a phase into smaller sub-phases based on allowable variance of CPI within a sub-phase also exists.

The proposed technique is implemented on simulators and on a native platform. Other advantages of phases in the context of timing analysis are also presented that include parallelized WCET analysis and estimation of remaining worst case execution time for a particular program run.